IN THE UNITED STATES DISTRICT COURT

FOR THE DISTRICT OF DELAWARE

DATA ENGINE TECHNOLOGIES	**§**	
LLC,	**§**	**C.A. No. 14-1115-LPS**
	§	
Plaintiff,	**§**	
	§	**JURY TRIAL DEMANDED**
v.	**§**	
	§	
GOOGLE INC.,	**§**	
	§	
Defendant.	**§**	

PLAINTIFF DATA ENGINE TECHNOLOGIES LLC'S RESPONSIVE CLAIM CONSTRUCTION BRIEF

Dated: November 20, 2015

Amir Alavi (admitted *pro hac vice*)
Demetrios Anaipakos (admitted *pro hac vice*)
Alisa A. Lipski (admitted *pro hac vice*)
Jamie A. Aycock (admitted *pro hac vice*)
Benjamin Foster (admitted *pro hac vice*)
Kenneth Young (admitted *pro hac vice*)
Timothy Shelby (admitted *pro hac vice*)
AHMAD, ZAVITSANOS, ANAIPAKOS,
ALAVI & MENSING, P.C.
1221 McKinney Street, Suite 3460
Houston, Texas 77010
Telephone: 713-655-1101
Facsimile: 713-655-0062
aalavi@azalaw.com
danaipakos@azalaw.com
alipski@azalaw.com
jamieaycock@azalaw.com
bfoster@azalaw.com
kyoung@azalaw.com
tshelby@azalaw.com

Brian E. Farnan (Bar No. 4089)
Michael J. Farnan (Bar No. 5165)
FARNAN LLP
919 N. Market St., 12th Floor
Wilmington, DE 19801
(302) 777-0300
bfarnan@farnanlaw.com
mfarnan@farnanlaw.com

Attorneys for Plaintiff
Data Engine Technologies LLC

TABLE OF CONTENTS

INTRODUCTION

Plaintiff Data Engine Technologies, LLC ("Data Engine") seeks to leave clear claim language in place. Defendant Google, Inc. ("Google") attempts to confuse the claim language by adding additional limitations in some instances from the preferred embodiment and in other instances just creating limitations wholesale. Google's attempts to rewrite the claims should be rejected.

TERMS FOR CONSTRUCTION

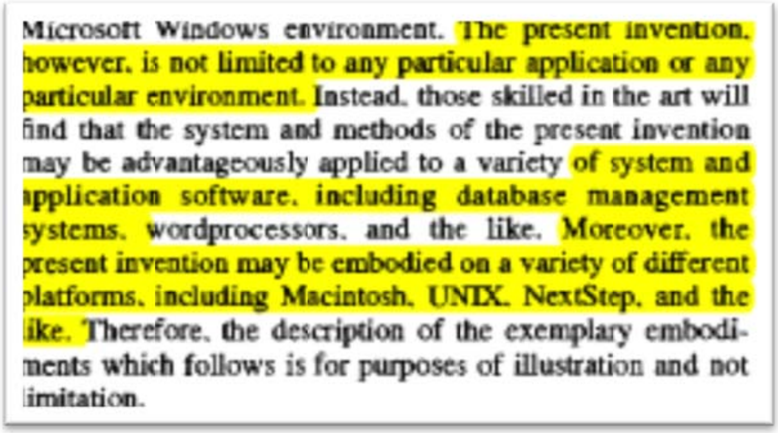
I. U.S. Patent No. 5,784,545

A. "Single disk file"

TERM	PLAINTIFF'S PROPOSED DEFINITION	DEFENDANT'S PROPOSED DEFINITION
Single disk file	Plain and ordinary meaning. No construction necessary.	Stored in a file on a single physical disk.

Google's argument regarding "single disk file" has its underpinnings in two improper assumptions. First, that the "mass storage" description found in a preferred embodiment of the patent must be imported into the claim language, and second, that "mass storage" in turn must be a "single physical disk."

First, Google contends that the specification only contains one embodiment, the one shown pictorially in Figure 1A. Google's Initial Claim Construction Brief, D.I. 87 ("Google Brief") at 15. However before its discussion of that embodiment the patentee explains that "As shown in FIG. 1A, the present invention ***may*** be embodied on a computer system such as the system 100." '545 at Col. 27, lines 27 (emphasis added). The use of "may" clearly denotes a possible embodiment, not an attempt by the patentee to limit the invention to a particular hardware configuration. The patentee goes on to recite:

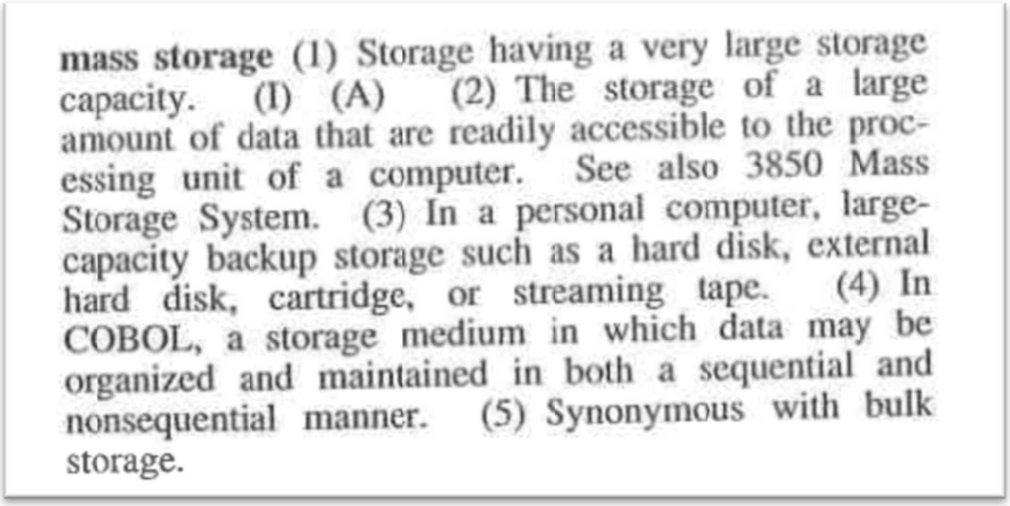


> Microsoft Windows environment. The present invention, however, is not limited to any particular application or any particular environment. Instead, those skilled in the art will find that the system and methods of the present invention may be advantageously applied to a variety of system and application software, including database management systems, wordprocessors, and the like. Moreover, the present invention may be embodied on a variety of different platforms, including Macintosh, UNIX, NextStep, and the like. Therefore, the description of the exemplary embodiments which follows is for purposes of illustration and not limitation.

‘545 at Col. 6, lines 1–12. This effectively rebuts any presumption that the invention is limited to a particular software or hardware configuration. Therefore, Google’s attempts to read a hardware limitation from figure 1A should be rejected.

However, even if the Court accepts that “single disk file” must mean “mass storage” in light of the hardware configuration in the preferred embodiment shown in Figure 1A it does not follow that Google’s construction is correct. The description of Figure 1A relating to storage is as follows: As “spreadsheet pages and one Graphs page, all of which are saved as a single disk file on the mass storage 107.” ‘545 at Col. 7:59–62. Yet Google does not seek a construction which includes the term “mass storage” instead, Google makes another leap, effectively construing “mass storage” to mean “single physical disk.” This construction is simply unsupportable. All the patentee says about the “mass storage” is, that one example of a mass storage is a hard disk. ‘545 at Col. 5, lines 32–33. One of skill in the art at the time of the invention would have been well aware of many methods of creating mass storage, many of which would not have been limited to a single physical hard disk, or a single mass storage. This was demonstrated by the extrinsic evidence cited in Data Engine’s opening brief. Data Engine

Brief at 7–8. Google proffers a definition from the IBM Dictionary of Computing, but skips the more relevant first and second definitions, instead choosing the definition which is specifically tied to “a personal computer”, a distinction not made in the patent:



mass storage (1) Storage having a very large storage capacity. (I) (A) (2) The storage of a large amount of data that are readily accessible to the processing unit of a computer. See also 3850 Mass Storage System. (3) In a personal computer, large-capacity backup storage such as a hard disk, external hard disk, cartridge, or streaming tape. (4) In COBOL, a storage medium in which data may be organized and maintained in both a sequential and nonsequential manner. (5) Synonymous with bulk storage.

IBM Dictionary of Computing, D.I. 91–1. Furthermore, Dr. Smedley, in his declaration, describes a variety of other storage techniques, which would have been well known in the art at the time of the patent. Smedley Declaration, at ¶¶ 7–10. In particular Dr. Smedley discusses distributed storage, a technique which predates the time of the patent in which file systems would share multiple physical hard disks. *Id.* In addition Dr. Smedley addresses a handful of mass storage solutions, apart from hard drives, that would have been known to one of skill in the art at the time of the invention. *Id*. at 10. In short, it would have been well known to one of skill in the art in 1992 that a “mass storage” was not limited to “a single physical disk.” Because “mass storage” cannot be construed to mean “single physical disk,” there is absolutely no justification, even under Google’s own understanding of Figure 1A, for limiting a single disk file to require a single physical disk.

II. U.S. Patent No. 6,282,551

A. "Storing said first and second pages of the plurality of cell matrices such that they appear to the user as being stored within a single file."

TERM	PLAINTIFF'S PROPOSED DEFINITION	DEFENDANT'S PROPOSED DEFINITION
Storing said first and second pages of the plurality of cell matrices such that they appear to the user as being stored within a single file	Storing said first and second pages of the plurality of cell matrices such that they are accessible by a single file name.	Indefinite.

Google suggests that the term: "appear to the user as being stored within a single file" is analogous to the canonical example of indefinite claim language: "aesthetically pleasing." But Google provides no reason for why these terms are analogous except their assertion that the terms are both "subjective." In contrast to the term "aesthetically pleasing" there are clear lines of delineation between whether something appears to be saved as one file, or appears to be saved to more than one file.[1]

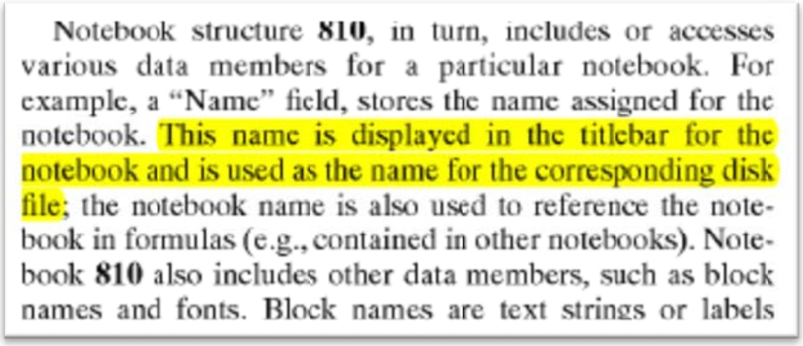
As discussed in Data Engine's opening brief, when courts have considered the word "appear," they have had little difficulty in discerning a limit to the term. Additionally, as discussed at length in the opening brief, the term "appear" is consistently used in the patent to mean "perceived by the user."

[1] DET notes that Google's own programmers use the term, see for example https://developers.google.com/web/updates/2014/11/Support-for-installable-web-apps-with-webapp-manifest-in-chrome-38-for-Android?hl=en describing "The Manifest for Web applications is a simple JSON file that gives you, the developer, the ability to control how your app *appears to the user*…"

Google argues that "appears" must be subjective because the limitation would "depend solely on the unrestrained, subjective opinion of a particular individual purportedly practicing the invention". Google Brief at 18. However, Google's brief provides no support for this proposition. There is no evidence that "[S]uch that they appear to the user" is neither a judgment call not a term of degree, and is in fact commonly used in the art, as described further below.

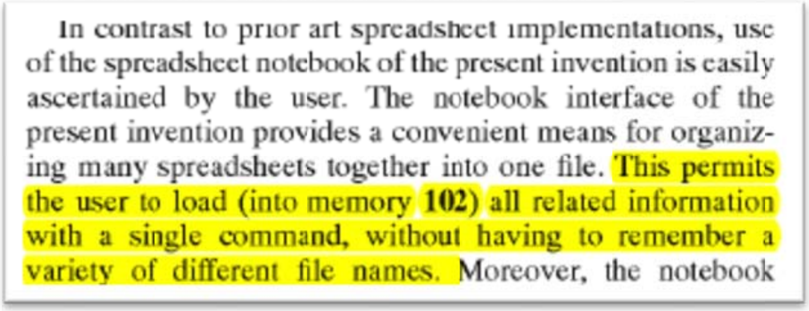
In the context of computing, the idea that something would "appear to be stored in a single file," is well understood. As stated by Dr. Smedley in his attached declaration, computer systems often conceal the underlying complexity of the implementation of a system from a user. Smedley Declaration, at ¶¶ 11–12. Here, what is claimed in the simple idea that regardless of what the underlying hardware implementation actually is, regardless of what tricks the system employs behind the scenes, the spreadsheet appears to the user to be stored in a single file. "Appear to the user as a single disk file" is distinct from "aesthetically pleasing," in that when a person views an object, based on their personal taste or preference they might choose to view the object as pleasing or not pleasing. However, the spreadsheet is either going to appear as being stored within a single disk file, or it is going to appear to be stored within multiple disk files. Google suggests no example, and Data Engine cannot think of one, where one user observing a system perceives the notebook as being stored within a single file, but another user perceives the notebook as being stored within multiple files.

Data Engine's proposed definition further clarifies "appears" by limiting "appears" to mean accessible by a single file name. There is no question that the patent discloses accessing a spreadsheet by a single file name:



Notebook structure **810**, in turn, includes or accesses various data members for a particular notebook. For example, a "Name" field, stores the name assigned for the notebook. This name is displayed in the titlebar for the notebook and is used as the name for the corresponding disk file; the notebook name is also used to reference the notebook in formulas (e.g., contained in other notebooks). Notebook **810** also includes other data members, such as block names and fonts. Block names are text strings or labels

Col. 15, lines 39–47. Additionally, the Patentee discloses the importance of having a single file name for each spreadsheet:



In contrast to prior art spreadsheet implementations, use of the spreadsheet notebook of the present invention is easily ascertained by the user. The notebook interface of the present invention provides a convenient means for organizing many spreadsheets together into one file. This permits the user to load (into memory **102**) all related information with a single command, without having to remember a variety of different file names. Moreover, the notebook

Col. 11, lines 27–34. Data Engine's proposal therefore embraces this key concept within the specification.

III. U.S. Patent No. 5,623,591

A. "linking"

TERM	PLAINTIFF'S PROPOSED DEFINITION	DEFENDANT'S PROPOSED DEFINITION
linking	Plain and ordinary meaning. No construction necessary.	Bidirectionally linking a property of a cell to a property of a user interface object such that when one object's property is changed, the other's property will be updated.

As Data Engine discussed in its opening claim construction brief, there is no dispute that claim 1 when read as a whole, requires bi-directional linking. That fact is inherent in the claim elements themselves and was made clear during the prosecution of the patent. In their opening briefs, Google and Data Engine discuss the same portions of the specification, but come to very different conclusions. As explained in Data Engine's opening brief, the examiner made it clear that it was the interpretation of claim 1 *as a whole* that the examiner described as "a bidirectional linking between the cell value property and the user interface object value property exists such that when one object's value property is changed, the other's value property will be updated." DETFH0002518. Google offers no reasoning for its insertion of this limitation of the claim as a whole into the "linking" term, an insertion which is not benign. Each claim term in a claim is presumed to have meaning, and Google cannot be allowed to cloud the claim limitations by inserting what is required by the second half of claim one into "linking".

B. "displaying said user interface object with a value of said property of the given cell object

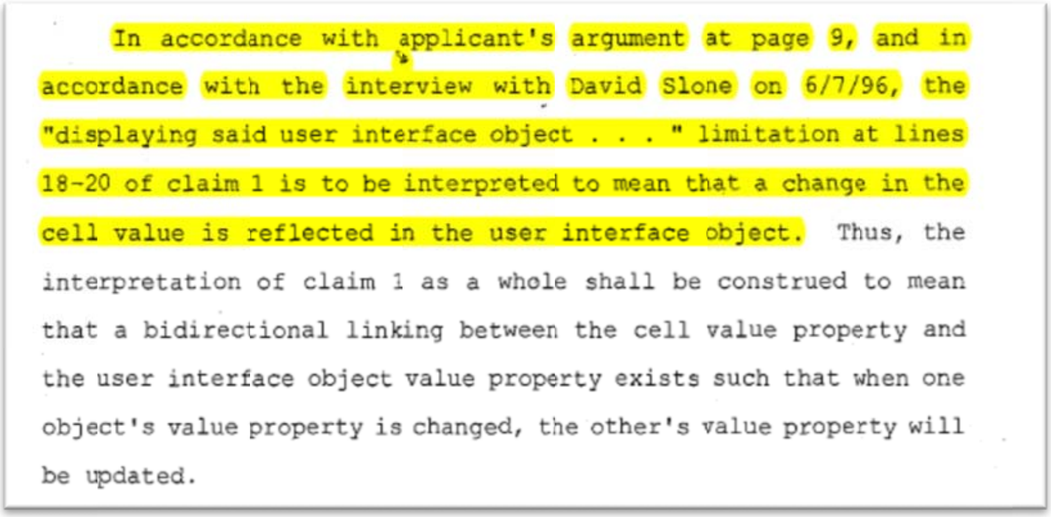
TERM	PLAINTIFF'S PROPOSED DEFINITION	DEFENDANT'S PROPOSED DEFINITION
Displaying said user interface object with a value of said value property of the given cell object	Plain and ordinary meaning. No construction necessary.	Automatically updating the value of said value property of said user interface object when the value of the value property of the corresponding cell is changed

Google claims its proposal for this term is in line with the definitions given in the file history, but the reality is that it is inexplicably different from those definitions. Shown below is Google's definition of this term, alongside both the Examiner's summary of an interview with the patentee and finally the Notice of Allowance itself. As shown, Google's definition is different from either of these references within the file history:

Google's Construction	Call Summary: (DETFH0002514)	Notice of Allowance: (DETFH0002516)
"Automatically updating the value of the value property of	"changing the value of 'said value property' in a	"a change in the cell value is reflected in the user interface

said user interface object when the value of the value property of the corresponding cell is changed.”	cell will update the corresponding value of the user interface object’s ‘value property’.	object.”

More complete text of the Notice of allowance is reproduced below:



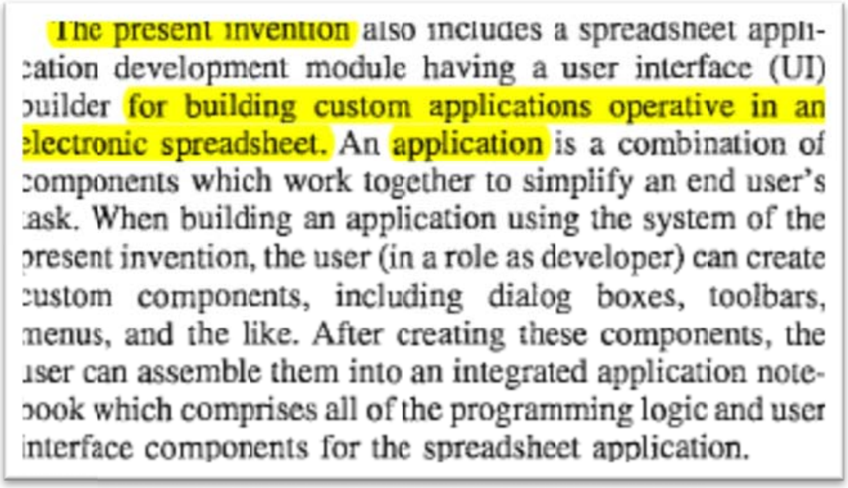
In accordance with applicant's argument at page 9, and in accordance with the interview with David Slone on 6/7/96, the "displaying said user interface object . . . " limitation at lines 18-20 of claim 1 is to be interpreted to mean that a change in the cell value is reflected in the user interface object. Thus, the interpretation of claim 1 as a whole shall be construed to mean that a bidirectional linking between the cell value property and the user interface object value property exists such that when one object's value property is changed, the other's value property will be updated.

The examiner’s construction make no mention of the arbitrary addition of the word “automatic” to the definition of “displaying” and retains the natural reading of the claim in context. It is not clear where Google gets the word “automatically” from in its definition. It is also not clear why the essential step of actually displaying the user interface object has been construed out of the claim term by Google. And it is certainly not clear why the convoluted language proposed by Google is clearer or more helpful than the examiners simple construction from the notice of allowance. While Data Engine believes that the plain and ordinary meaning of the term n is both more clear and more consistent with the specification than Google’s proposed definition, it is not opposed, as it argued in its opening brief, to entering a construction in line with the examiners construction, such as “displaying said user interface object such that a change in the cell value is reflected in the user interface object.”

C. “end-user input”

TERM	PLAINTIFF’S PROPOSED DEFINITION	DEFENDANT’S PROPOSED DEFINITION
End-user input	Input by a person who is running a custom application operative in an electronic spreadsheet to perform tasks	Input by a person who is running an application operative in an electronic spreadsheet to perform tasks

Google does not explain why it believes that “completed application” will be more clear for a juror than “custom application operative in an electronic spreadsheet.” In fact, it is Data Engine’s understanding of the specification that these two phrases mean exactly the same thing:



The present invention also includes a spreadsheet application development module having a user interface (UI) builder for building custom applications operative in an electronic spreadsheet. An application is a combination of components which work together to simplify an end user’s task. When building an application using the system of the present invention, the user (in a role as developer) can create custom components, including dialog boxes, toolbars, menus, and the like. After creating these components, the user can assemble them into an integrated application notebook which comprises all of the programming logic and user interface components for the spreadsheet application.

‘591 Patent at Col. 9, lines 4–20. As shown in the citation above the present invention is for building custom applications operative in an electronic spreadsheet. This lengthy descriptive phrase is often shortened in the specification just to application, as can be seen from the very next line of the specification, which recites that “an application is . . . .”

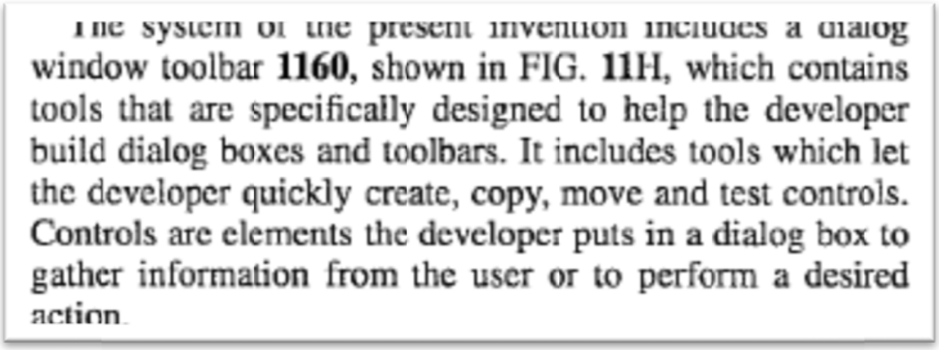
The problem with Google’s definition—besides its inconsistency with the definition of “user,”—is that it would require further construction of the term “application” which is fairly generic in the context of software. Since the patent clearly relates to building applications within spreadsheets, the Court should adopt Data Engine’s more precise and descriptive definition of the type of applications creatable by the methods of the ‘591.

D. “in response to first user input, generating a user interface object/in response to third user input, generating a user interface object” / “receiving first user input for generating a user interface control of a plurality of different types”

TERM	**PLAINTIFF’S PROPOSED DEFINITION**	**DEFENDANT’S PROPOSED DEFINITION**
In response to first user input, generating a user interface object Claim 1 In response to third user input, generating a user interface object	Plain and ordinary meaning. No construction necessary.	Generating a user interface object of a predefined type distinct from cell objects in response to a user selecting a user interface object type in a graphical user interface
Claim 3 Receiving first user input for generating a user interface control of a plurality of different types for receiving end-user input	Plain and ordinary meaning. No construction necessary.	Receiving user input selecting a user interface control type in a graphical user interface

Google correctly summarizes the dispute with respect to these terms, namely whether “generating a user interface object” requires “selecting a user interface object type in a graphical user interface.” There is no doubt that the preferred embodiment of the ‘591 patent is illustrated as a graphical user interface. Google argues that because the patentee states that “The system of the invention includes a dialog window toolbar” that that toolbar must be the only way to receive user input for generating a user interface object. This is rebutted by the evidence that DET has discussed in its opening claim construction brief, in which the patentee indicates that he is not limiting himself to any particular environment, much less a Graphical User interface specific environment.

Google's argument rests on the assumption that the following language should be used to construe "generating a user interface object." However, those words do not appear in this section of specification:



The system of the present invention includes a dialog window toolbar **1160**, shown in FIG. **11**H, which contains tools that are specifically designed to help the developer build dialog boxes and toolbars. It includes tools which let the developer quickly create, copy, move and test controls. Controls are elements the developer puts in a dialog box to gather information from the user or to perform a desired action.

'591 at Col. 29, lines 31–38. There is simply language here that would explicitly limit the generation of user interface objects to the use of a toolbar, nor is there a disclosure here that the toolbar must necessarily be a graphical user interface. Nor is there anything in the specification that would require that this "toolbar" even be a graphical user interface.

Perhaps most importantly, while the system described in detail in the specification included a toolbar, there is nothing that limits the asserted claims to that particular toolbar. Even Google's own citation supports this argument, Google cites to language in the specification related to: "The system of the present invention" However, the term Google seeks to construe with this language is found in the method claim (claim 1), not the system claim (claim 7). Those method claims, by their very nature, are not limited to any specific system.

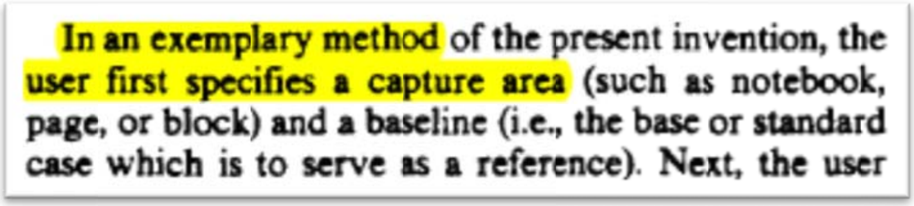
Google also claims that the prosecution history supports its definition. However, the rejection cited by Google does not support its argument. Google Brief at 25. There Google discusses the Kanai reference which was traversed by pointing out that the invention requires the user interface controls to be user created. DETFH0002412. There is no dispute that the '591 requires user created user interface objects. The dispute is that Google seeks to add the additional limitation that these user interface objects be generated only by "selecting a user interface object type in a graphical user interface." When responding to this rejection, the patentee did not characterize his invention as limited to the user creation of user interface elements through a graphical user interface and so Google's argument regarding the Kanai rejection is misplaced.

IV. U.S. Patent No. 5,303,146

A. "Specifying a base set of information cells"

TERM	PLAINTIFF'S PROPOSED DEFINITION	DEFENDANT'S PROPOSED DEFINITION
Specifying a base set of information cells	Plain and ordinary meaning. No construction necessary.	User selecting a set of cells in an open notebook as a base set from which user defined scenarios are created

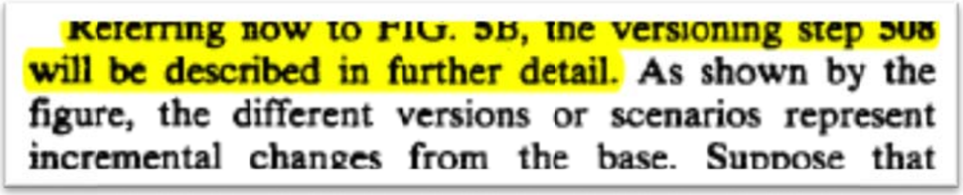
Google suggests that "specifying a base set of information cells" necessarily means "user selecting a set of cells." However, this is an attempt by a defendant to import limitations from the preferred embodiment of the patent into an independent claim, when the preferred embodiment is explicitly claimed in the more limited dependent claims. The term in the patent for a user selected set of cells is "capture area." As explained in Data Engine's opening brief this concept is claimed in claim 2 and its dependents, none of which are asserted in this case. Google's own citations to the specification reconfirm again and again that the concept of a "user selecting a set of cells" is described in the patent with reference to the "user selecting a capture area", the phrase found in the dependent claims. Google's first citation is to Col. 3 lines 3–6 and is reproduced below:



> In an exemplary method of the present invention, the user first specifies a capture area (such as notebook, page, or block) and a baseline (i.e., the base or standard case which is to serve as a reference). Next, the user

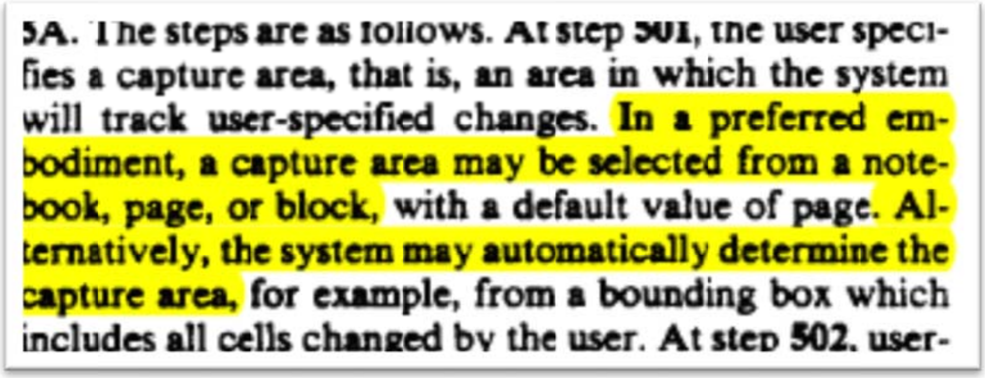
The use of the phrase "[i]n an exemplary method" by the specification leaves no ambiguity that what follows is a description of a preferred embodiment of the patent. Furthermore, the embodiment discussed here references "specifying a capture area" which is the exact language from dependent claim 2.

Likewise, Google's reliance on Figure 5B is also misplaced. Figure 5B relates to the "versioning step" which relates to elements 1(b) and 1(c) of claim 1:



> Referring now to FIG. 5B, the versioning step 508 will be described in further detail. As shown by the figure, the different versions or scenarios represent incremental changes from the base. Suppose that

Col. 13, lines 18–19. The specifying step, the focus of this claim construction dispute, is found in element 1(a) of claim 1 which is described with reference to Fig 5A:



> 5A. The steps are as follows. At step 501, the user specifies a capture area, that is, an area in which the system will track user-specified changes. In a preferred embodiment, a capture area may be selected from a notebook, page, or block, with a default value of page. Alternatively, the system may automatically determine the capture area, for example, from a bounding box which includes all cells changed by the user. At step 502, user-

Col. 12, lines 52–59. As shown, in a preferred embodiment, the capture area is selected. However, "Alternatively, the system may automatically determine the capture area." Limiting "specifying" to "user specifying" reads this embodiment—clearly described in the specification—out of the claims.

Furthermore, Data Engine is not attempting to write "specifying" out of the claim. Data Engine's theory is simple. When a user creates a new spreadsheet, he or she is specifying to the system an entirely new range of cells, namely the content of that spreadsheet, should now be tracked. This does not eliminate the work of the word "specified" instead Data Engine merely seeks to avoid artificial limitations, not contemplated by the patent, on how exactly the system decides what cells to track. Google's citation to the preamble of the claim is irrelevant. There is no dispute that the spreadsheet system is for "modeling user-specified information" but "information" in the specification is typically used in the patent to mean the data in spreadsheets, for example: "For one, electronic spreadsheets are much larger (i.e. hold more information) that their paper counterparts" Col 2, lines 10–11. There is no dispute that the methods of the present invention track user–specified information (i.e. track data the user types into the spreadsheet) but that fact is irrelevant to the claim construction issue before the Court.

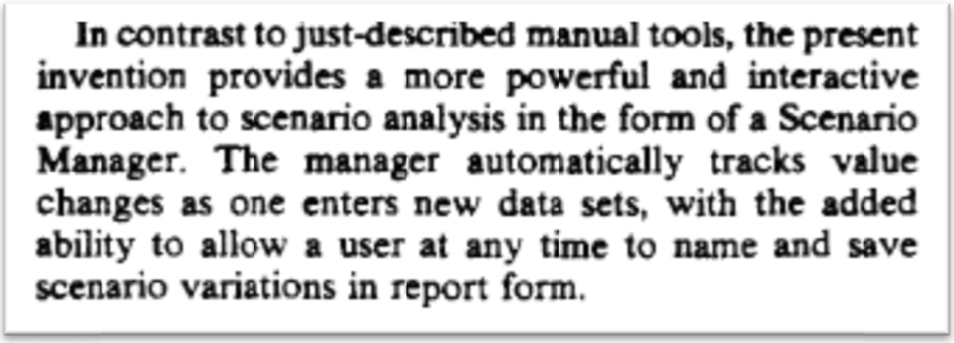
Finally, Google offers no support whatsoever for its proposed "in an open notebook" limitation. If Data Engine properly understands this added limitation, it is intended to require that the spreadsheet be open before specification begins. In fact the word "open" does not appear once in the patent.

B. The "version terms"

TERM	PLAINTIFF'S PROPOSED DEFINITION	DEFENDANT'S PROPOSED DEFINITION
Different version	Plain and ordinary meaning. No construction necessary.	User defined scenarios made by modifying the base version
New version		New user defined scenario made by modifying the base version
Base version		A single user defined reference version which is modified to

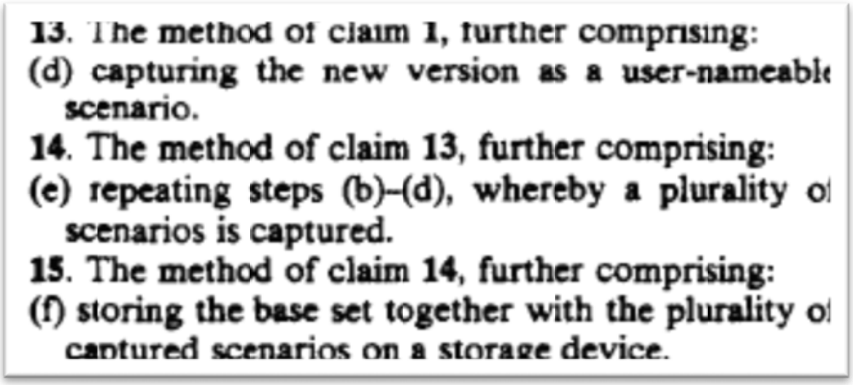
TERM	PLAINTIFF’S PROPOSED DEFINITION	DEFENDANT’S PROPOSED DEFINITION
		create a new scenario

Google provides essentially no support for its Google’s constructions, citing to a single piece of the specification:



In contrast to just-described manual tools, the present invention provides a more powerful and interactive approach to scenario analysis in the form of a Scenario Manager. The manager automatically tracks value changes as one enters new data sets, with the added ability to allow a user at any time to name and save scenario variations in report form.

Col. 7, lines 62–68. This citation does not even mention the term “version” and relates to an aspect of the invention that is clearly claimed in the dependent claims:



13. The method of claim 1, further comprising:
(d) capturing the new version as a user-nameabl scenario.
14. The method of claim 13, further comprising:
(e) repeating steps (b)–(d), whereby a plurality o scenarios is captured.
15. The method of claim 14, further comprising:
(f) storing the base set together with the plurality o captured scenarios on a storage device.

Col. 15, lines 3–11. Once again, Google is attempting to read limitations, claimed in dependent claims, into independent claims.

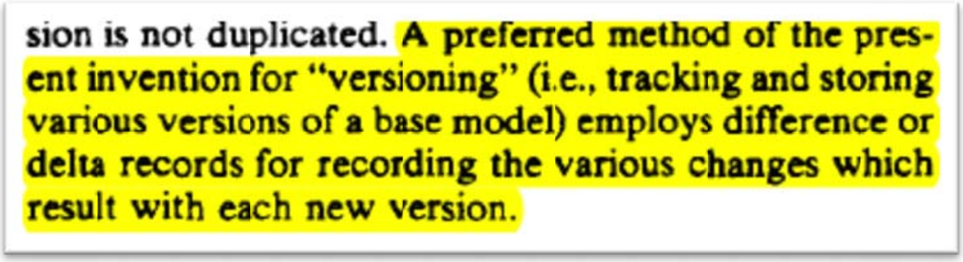
Furthermore, Google provides no way to reconcile the inconsistencies of its definitions, and no description of why “different versions” in the preamble should be limiting.

Finally, Google's arguments characterizing Data Engine's position in recent motion practice are irrelevant. Google is citing to language where Data Engine describes how "modifying at least one information cell" is performed by the Google system. That statement has nothing to do with the terms Google selected for construction, or the proper meaning of the term according to the intrinsic and extrinsic evidence available to a person of ordinary skill at the time of the invention.

C. "maintaining a new version by storing additional information for only those portions determined to have changed"

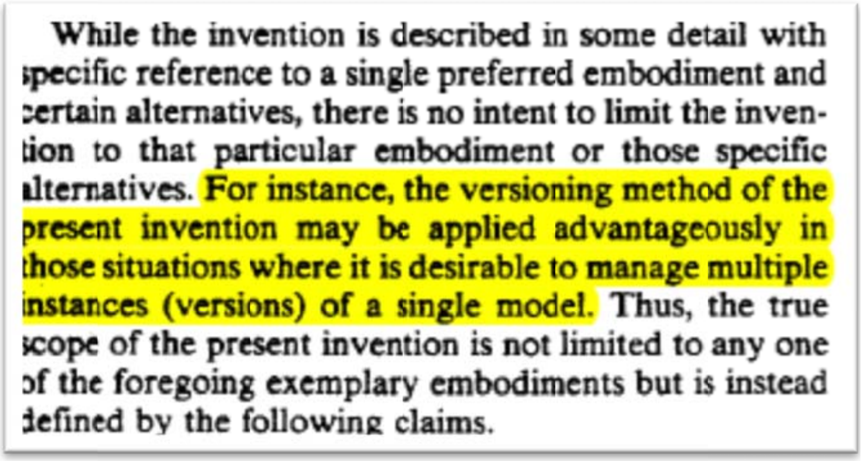
TERM	PLAINTIFF'S PROPOSED DEFINITION	DEFENDANT'S PROPOSED DEFINITION
Maintaining the new version by storing additional information for only those portions determined to have changed	Plain and ordinary meaning. No construction necessary.	Maintaining the new version by storing only portions of the new version which have changed when compared against the base version

Google's construction is designed to limit the scope of the claims by requiring that all versions be stored by the system as comparisons against only the base version. (as opposed to for example, the most recent prior version). The claim language makes clear that the invention is directed towards storing information only related to the changes, rather than storing information for portions that have not changed. In the Summary of the Invention, the patentee makes clear that there are different methods for tracking and that the delta method that Google seeks to import into Claim 1 is only a preferred embodiment, see '146 patent, col. 3, lines 34–38:



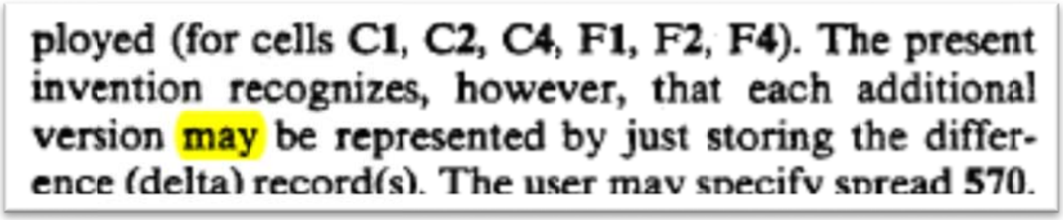
sion is not duplicated. A preferred method of the present invention for "versioning" (i.e., tracking and storing various versions of a base model) employs difference or delta records for recording the various changes which result with each new version.

Additionally, the patentee specifically included variations on the method of versioning in its recital of the invention not being limited to a single embodiment, see '146 patent, col. 13, lines 58–67:



While the invention is described in some detail with specific reference to a single preferred embodiment and certain alternatives, there is no intent to limit the invention to that particular embodiment or those specific alternatives. For instance, the versioning method of the present invention may be applied advantageously in those situations where it is desirable to manage multiple instances (versions) of a single model. Thus, the true scope of the present invention is not limited to any one of the foregoing exemplary embodiments but is instead defined by the following claims.

Google seeks to make this "preferred method" for "versioning" a required limitation of the claim. The portions of the specification that Google cites are consistent with the understanding that the "delta" method is just one way of providing versioning. See, '146 patent, Col. 13, lines 24 – 27:



ployed (for cells C1, C2, C4, F1, F2, F4). The present invention recognizes, however, that each additional version may be represented by just storing the difference (delta) record(s). The user may specify spread 570.

Because the patentee has not clearly limited the patent to a single method for versioning the "Maintaining" claim element should not be construed to require it.

CONCLUSION

For the foregoing reasons, and the reasons originally set forth in Data Engine's Opening Brief, Data Engine respectfully requests that the Court adopt its proposed claim constructions because its proposed constructions more closely adhere to the language set out in the patents themselves and represent how these terms would be understood by a person of ordinary skill in the art.

Dated: November 20, 2015

Respectfully submitted,

FARNAN LLP

/s/ Brian E. Farnan
Brian E. Farnan (Bar No. 4089)
Michael J. Farnan (Bar No. 5165)
919 N. Market St., 12th Floor
Wilmington, DE 19801
(302) 777-0300
bfarnan@farnanlaw.com
mfarnan@farnanlaw.com

Amir Alavi (admitted *pro hac vice*)
Demetrios Anaipakos (admitted *pro hac vice*)
Alisa A. Lipski (admitted *pro hac vice*)
Jamie A. Aycock (admitted *pro hac vice*)
Benjamin Foster (admitted *pro hac vice*)
Kenneth Young (admitted *pro hac vice*)
Timothy Shelby (admitted *pro hac vice*)
AHMAD, ZAVITSANOS, ANAIPAKOS,
ALAVI & MENSING, P.C.
1221 McKinney Street, Suite 3460
Houston, Texas 77010
Telephone: 713-655-1101
Facsimile: 713-655-0062
aalavi@azalaw.com
danaipakos@azalaw.com

alipski@azalaw.com
jamieaycock@azalaw.com
bfoster@azalaw.com
kyoung@azalaw.com
tshelby@azalaw.com

Attorneys for Plaintiff
Data Engine Technologies LLC